The Design and Architecture of MAQAOPROFILE: Assembly Instrumentation Module

L. Djoudi, D. Barthou, O. Tomaz, A. Charif-Rubial, J.T. Acquaviva and W. Jalby

University of Versailles St-Quentin-en-Yvelines, CEA

FRANCE

Outline

- Goals of instrumentation
- Performance tuning platform: MAQAO
- Assembly instrumentation features
- Profiling Scenarii
  - Block level profiling: Hot Paths
  - Loop Level Profiling
  - Instruction Level Profiling
- Performance Advisor
- Conclusion

Goals of Instrumentation

Where ?

- Instrumentation assembly code
- Instrumentation can focus at several levels of granularity
- On restructured code
- Instrumentation does not disturb the original code

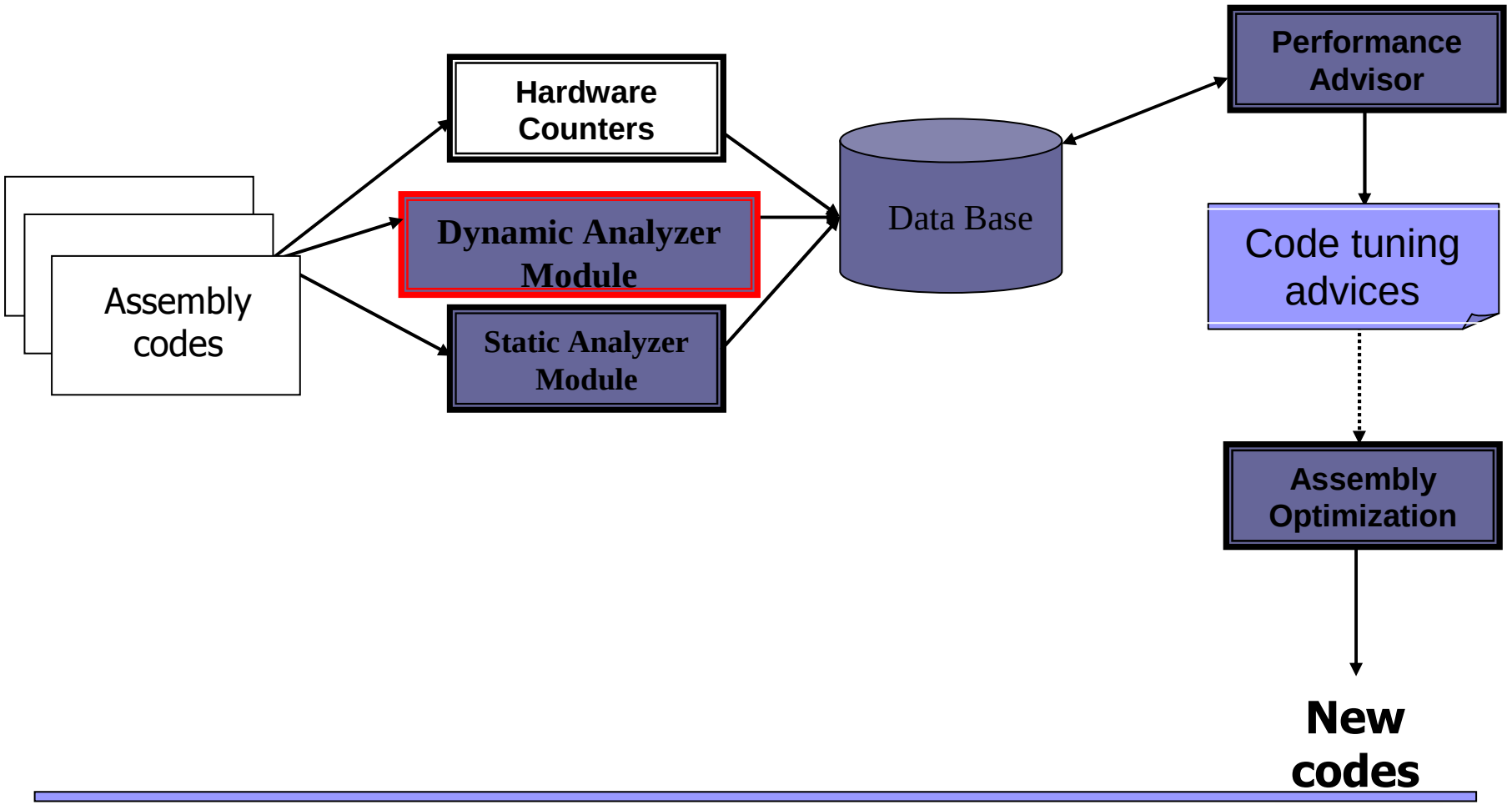
What ?

- Time and value profiling

How ?

- Interpretation with performance advisor

Performance Tuning platform(1)



Performance Tuning platform(2)

Improving code quality

- Find the parts of code to be optimized
- Detection of inefficient code patterns
- Check the behaviour during execution
- Report results based on static and dynamic analyses

MAQAO Functionalities

- Navigate through assembly and source codes
- Handle large application codes
- Enable user to drive analysis by scripts
- Designed for interactive and batch mode

Tuning approach

- Profile (Hardware counters)
- Check for known bad code patterns on hot functions
- Compute bounds from the assembly
  - Memory bound
  - Computation bound
- Compare bounds and compiler schedule:
  - mismatching: code quality problem. Check compiler flag (black belt, source modification)
- Compare profiling info. and compiler schedule:
  - mismatching: dynamic phenomenon. Use hardware counters to solve the issue
- Value Profiling:
  - Function specialization
  - Tuned pipeline depth
- Optimization
- Repeat until performance is at the top

Instrumentation principle

- Insertion of assembly probes
- Where do we inject these probes ?
- Data collected by probes
  - Builds execution time profile
  - Value profiling
  - Numerous optimization opportunities.

Features of Instrumentation

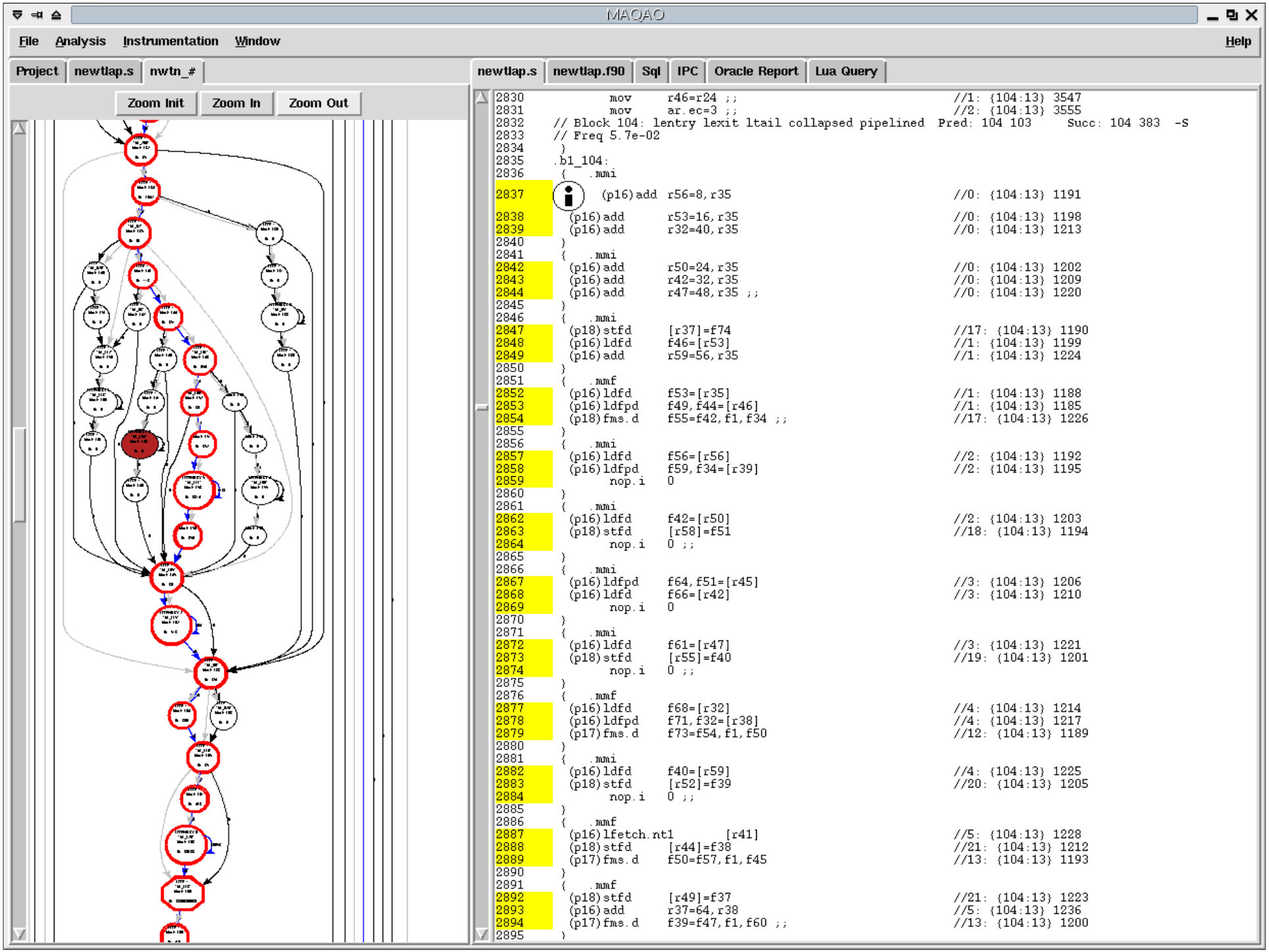
- Precise Instrumentation
  - Program/Blocks
    - Execution frequency
    - Execution time
  - Functions
    - Execution time
  - Loops
    - Value profiling
    - Execution time
  - Instructions
    - Value profiling
- Existing scenarii
  - Hotpaths
  - Loops
  - Value profiling of address register
- More scenarii will be developped

Related Works

- Vtune, Shark, …
- PIN, ATOM
- HPCVIEW

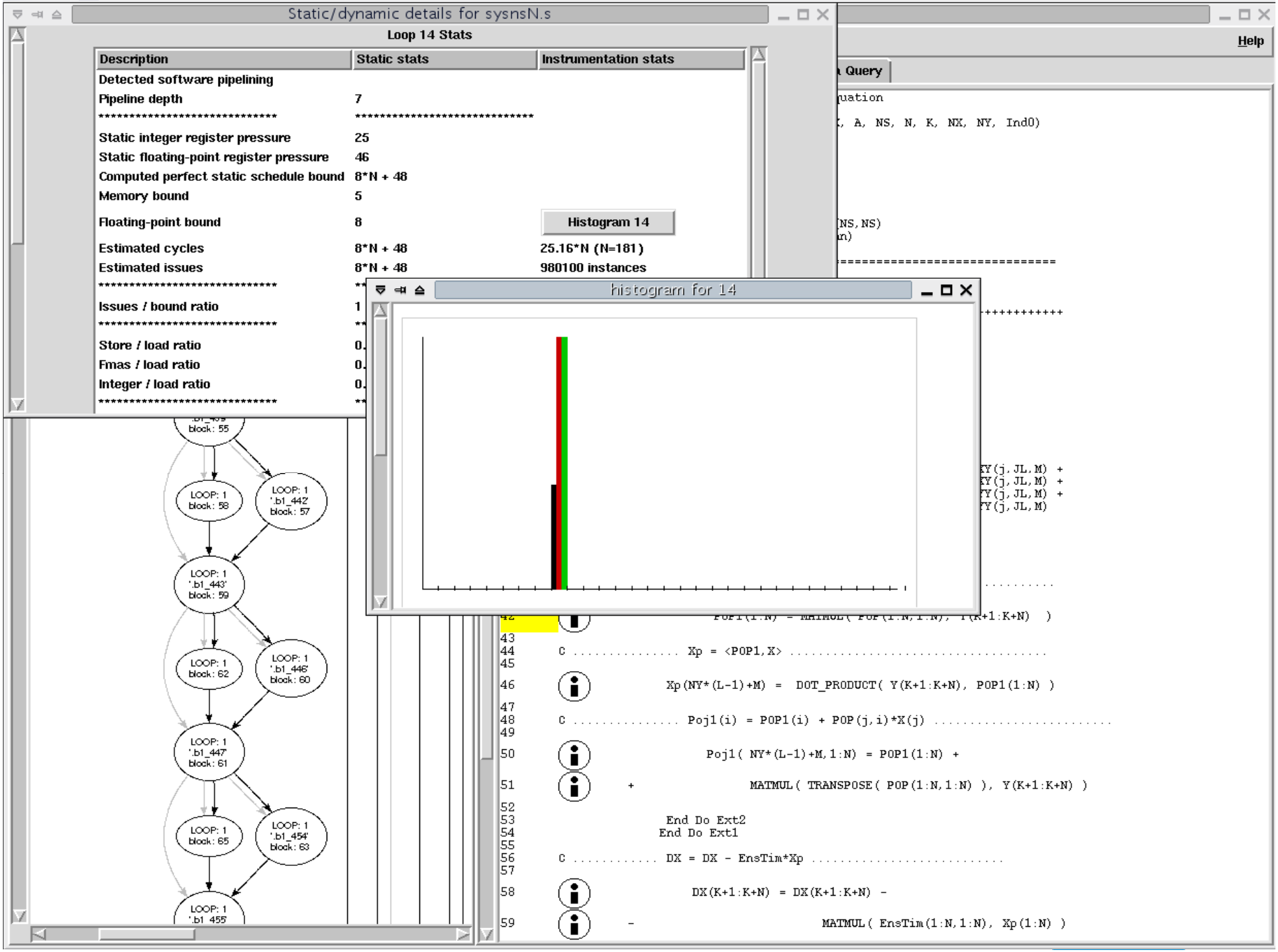
Block level profiling: Hot Paths

- Path of the most frequently executed blocks
- Hotpath
  - Frequency computed by probes for each block
  - Detection based on Bellman algorithm


MAQAO
File
Analysis
Instrumentation
Window
Help
Project
newtlap.s
nwtn_#
Zoom Init
Zoom In
Zoom Out
newtlap.s
newtlap.f90
Sql
IPC
Oracle Report
Lua Query
2830 mov r46=r24 ;; //1: {104:13} 3547
2831 mov ar.ec=3 ;; //2: {104:13} 3555
2832 // Block 104: lentry lexit ltail collapsed pipelined Pred: 104 103 Succ: 104 383 -S
2833 // Freq 5.7e-02
2834 }
2835 .b1_104:
2836 { .mmi
2837 (p16)add r56=8,r35 //0: {104:13} 1191
2838 (p16)add r53=16,r35 //0: {104:13} 1198
2839 (p16)add r32=40,r35 //0: {104:13} 1213
2840 }
2841 { .mmi
2842 (p16)add r50=24,r35 //0: {104:13} 1202
2843 (p16)add r42=32,r35 //0: {104:13} 1209
2844 (p16)add r47=48,r35 ;; //0: {104:13} 1220
2845 }
2846 { .mmi
2847 (p18)stfd [r37]=f74 //17: {104:13} 1190
2848 (p16)ldfd f46=[r53] //1: {104:13} 1199
2849 (p16)add r59=56,r35 //1: {104:13} 1224
2850 }
2851 { .mmf
2852 (p16)ldfd f53=[r35] //1: {104:13} 1188
2853 (p16)ldfpd f49,f44=[r46] //1: {104:13} 1185
2854 (p18)fms.d f55=f42,f1,f34 ;; //17: {104:13} 1226
2855 }
2856 { .mmi
2857 (p16)ldfd f56=[r56] //2: {104:13} 1192
2858 (p16)ldfpd f59,f34=[r39] //2: {104:13} 1195
2859 nop.i 0
2860 }
2861 { .mmi
2862 (p16)ldfd f42=[r50] //2: {104:13} 1203
2863 (p18)stfd [r58]=f51 //18: {104:13} 1194
2864 nop.i 0 ;;
2865 }
2866 { .mmi
2867 (p16)ldfpd f64,f51=[r45] //3: {104:13} 1206
2868 (p16)ldfd f66=[r42] //3: {104:13} 1210
2869 nop.i 0
2870 }
2871 { .mmi
2872 (p16)ldfd f61=[r47] //3: {104:13} 1221
2873 (p18)stfd [r55]=f40 //19: {104:13} 1201
2874 nop.i 0 ;;
2875 }
2876 { .mmf
2877 (p16)ldfd f68=[r32] //4: {104:13} 1214
2878 (p16)ldfpd f71,f32=[r38] //4: {104:13} 1217
2879 (p17)fms.d f73=f54,f1,f50 //12: {104:13} 1189
2880 }
2881 { .mmi
2882 (p16)ldfd f40=[r59] //4: {104:13} 1225
2883 (p18)stfd [r52]=f39 //20: {104:13} 1205
2884 nop.i 0 ;;
2885 }
2886 { .mmf
2887 (p16)lfetch.nt1 [r41] //5: {104:13} 1228
2888 (p18)stfd [r44]=f38 //21: {104:13} 1212
2889 (p17)fms.d f50=f57,f1,f45 //13: {104:13} 1193
2890 }
2891 { .mmf
2892 (p18)stfd [r49]=f37 //21: {104:13} 1223
2893 (p16)add r37=64,r38 //5: {104:13} 1236
2894 (p17)fms.d f39=f47,f1,f60 ;; //13: {104:13} 1200
2895 }

Loop Level Profiling

- For inner loops
- Profiling for all loop versions
- Data collected:
  - whether the loop is executed
  - Average execution time/iteration
  - Distribution of loop trip count
  - Distribution of cycles/iteration
- Provides hot/short loops
- Opportunities for specific optimization
  - Softwarepipeline tuning depth
  - Prefetch distance


Static/dynamic details for sysnsN.s
Loop 14 Stats
Description
Static stats
Instrumentation stats
Detected software pipelining
Pipeline depth
7
Static integer register pressure
25
Static floating-point register pressure
46
Computed perfect static schedule bound
8*N + 48
Memory bound
5
Floating-point bound
8
Histogram 14
Estimated cycles
8*N + 48
25.16*N (N=181)
Estimated issues
8*N + 48
980100 instances
Issues / bound ratio
1
Store / load ratio
Fmas / load ratio
Integer / load ratio
histogram for 14
Help
Query
LOOP: 1
block: 55
LOOP: 1
block: 58
LOOP: 1
'.b1_442'
block: 57
LOOP: 1
'.b1_443'
block: 59
LOOP: 1
block: 62
LOOP: 1
'.b1_446'
block: 60
LOOP: 1
'.b1_447'
block: 61
LOOP: 1
block: 65
LOOP: 1
'.b1_454'
block: 63
LOOP: 1
'.b1_455'
C ..... Xp = <POP1,X> .....
Xp(NY*(L-1)+M) = DOT_PRODUCT(Y(K+1:K+N), POP1(1:N))
C ..... Poj1(i) = POP1(i) + POP(j,i)*X(j) .....
Poj1(NY*(L-1)+M,1:N) = POP1(1:N) +
+ MATMUL(TRANSPOSE(POP(1:N,1:N)), Y(K+1:K+N))
End Do Ext2
End Do Ext1
C ..... DX = DX - EnsTim*Xp .....
DX(K+1:K+N) = DX(K+1:K+N) -
- MATMUL(EnsTim(1:N,1:N), Xp(1:N))

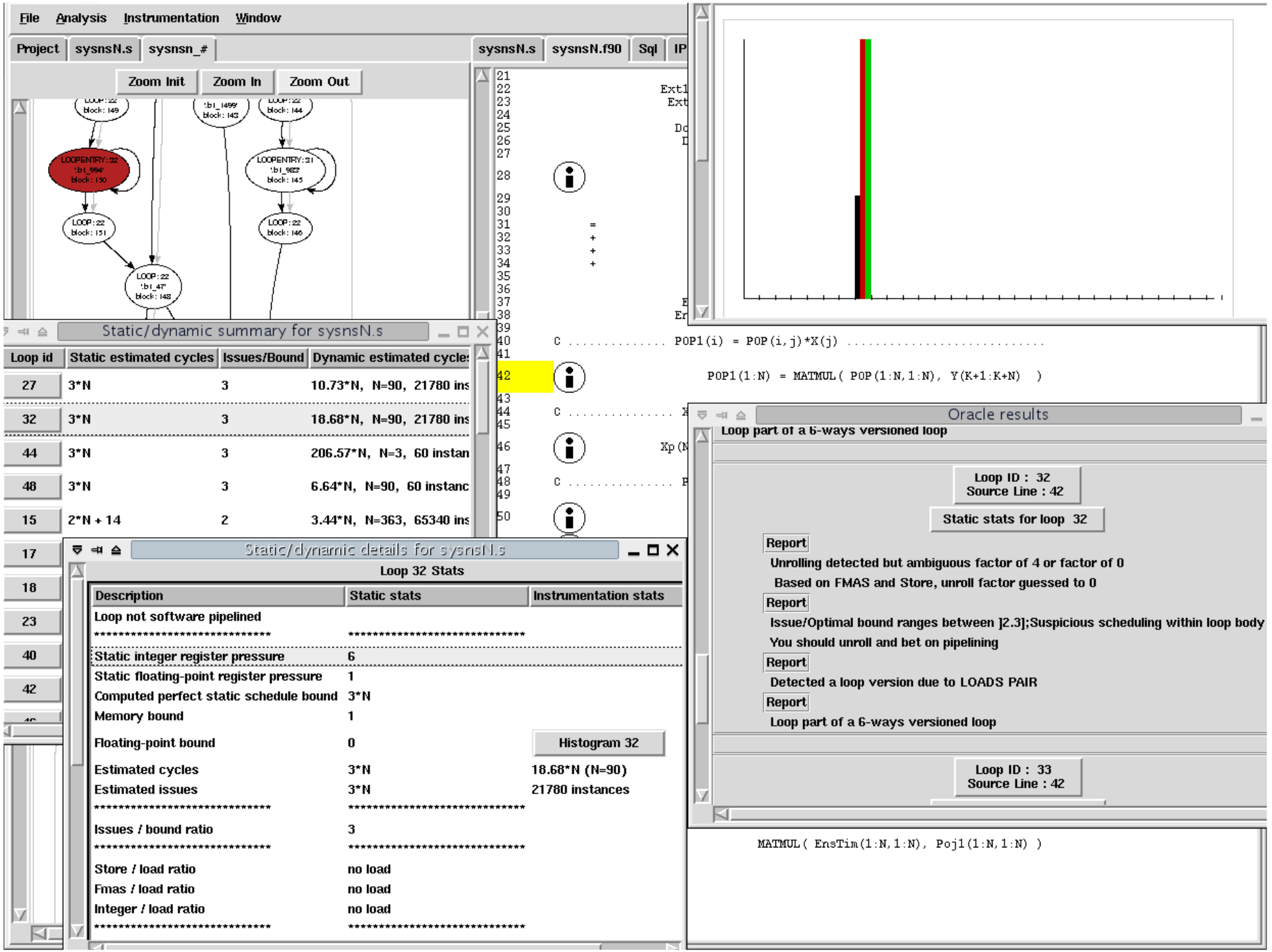
Profiling of register values

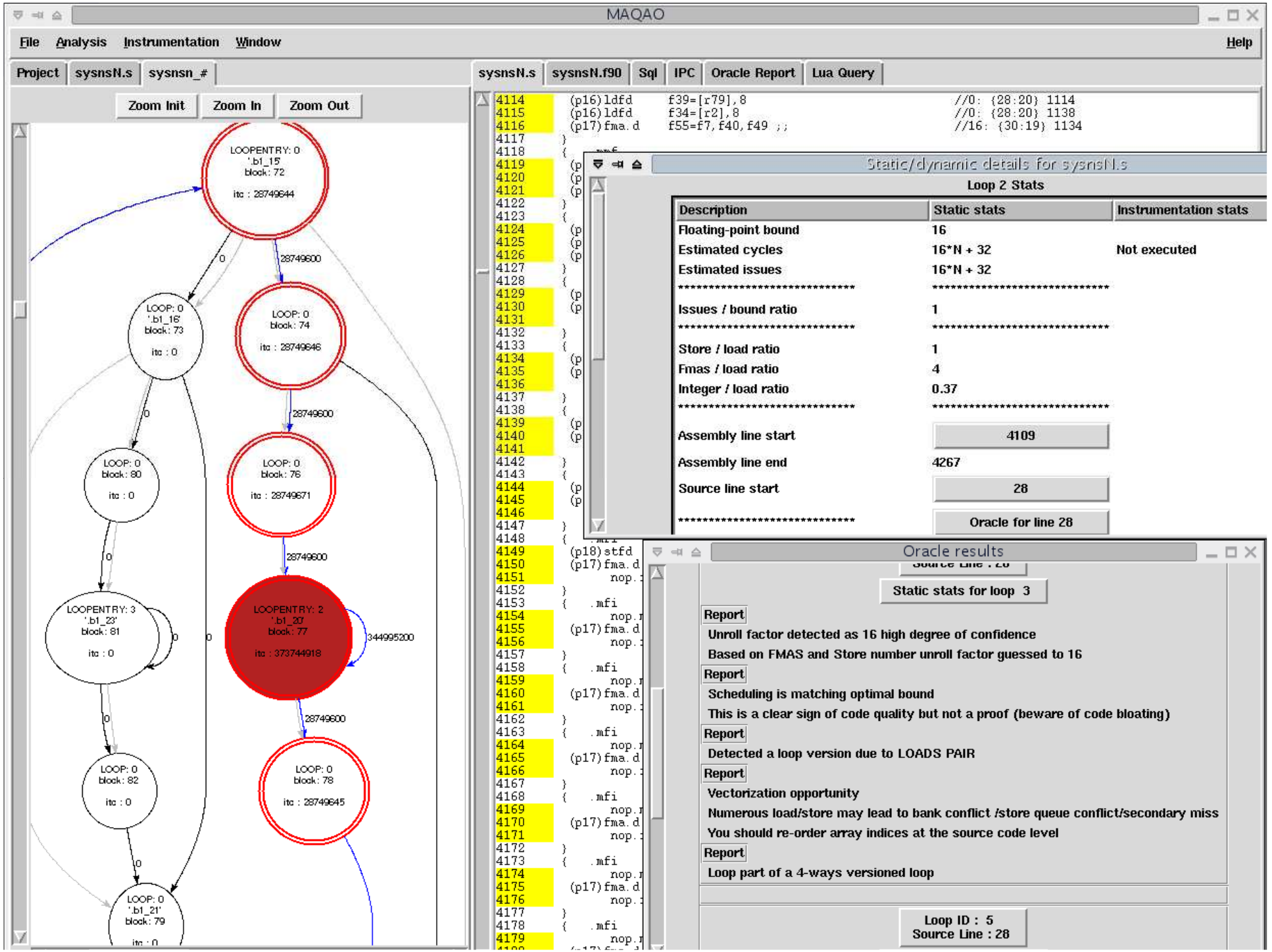
Capture values of selected registers

- Functions parameters:
  - Detect opportunities for specialization
- Address registers
  - Used to compute prefetch distances
  - Used to detect bank conflicts

Performance Advisor

- How to give, manage and extract important tuning information
  - Static Analysis
  - Hierarchical Reporting Approach
- Built over a set of rules and metrics
- Help user to deal with complex architecture
- Provides an optimization report


File Analysis Instrumentation Window
Project sysnsN.s sysnsn_#
Zoom Init Zoom In Zoom Out
sysnsN.s sysnsN.f90 Sql IP
Static/dynamic summary for sysnsN.s
Loop id | Static estimated cycles | Issues/Bound | Dynamic estimated cycles
27 | 3*N | 3 | 10.73*N, N=90, 21780 ins
32 | 3*N | 3 | 18.68*N, N=90, 21780 ins
44 | 3*N | 3 | 206.57*N, N=3, 60 instan
48 | 3*N | 3 | 6.64*N, N=90, 60 instanc
15 | 2*N + 14 | 2 | 3.44*N, N=363, 65340 ins
17
18
23
40
42
POP1(i) = POP(i,j)*X(j)
POP1(1:N) = MATMUL(POP(1:N,1:N), Y(K+1:K+N))
Static/dynamic details for sysnsN.s
Loop 32 Stats
Description | Static stats | Instrumentation stats
Loop not software pipelined
Static integer register pressure | 6
Static floating-point register pressure | 1
Computed perfect static schedule bound | 3*N
Memory bound | 1
Floating-point bound | 0
Histogram 32
Estimated cycles | 3*N | 18.68*N (N=90)
Estimated issues | 3*N | 21780 instances
Issues / bound ratio | 3
Store / load ratio | no load
Fmas / load ratio | no load
Integer / load ratio | no load
Oracle results
Loop part of a 6-ways versioned loop
Loop ID : 32
Source Line : 42
Static stats for loop 32
Report
Unrolling detected but ambiguous factor of 4 or factor of 0
Based on FMAS and Store, unroll factor guessed to 0
Report
Issue/Optimal bound ranges between]2.3];Suspicious scheduling within loop body
You should unroll and bet on pipelining
Report
Detected a loop version due to LOADS PAIR
Report
Loop part of a 6-ways versioned loop
Loop ID : 33
Source Line : 42
MATMUL(EnsTim(1:N,1:N), Poj1(1:N,1:N))


MAQAO
File
Analysis
Instrumentation
Window
Help
Project
sysnsN.s
sysnsn_#
Zoom Init
Zoom In
Zoom Out
LOOPENTRY: 0
'.b1_15'
block: 72
itc : 28749644
LOOP: 0
'.b1_16'
block: 73
itc : 0
LOOP: 0
block: 74
itc : 28749646
LOOP: 0
block: 80
itc : 0
LOOP: 0
block: 76
itc : 28749671
LOOPENTRY: 3
'.b1_23'
block: 81
itc : 0
LOOPENTRY: 2
'.b1_20'
block: 77
itc : 373744918
LOOP: 0
block: 82
itc : 0
LOOP: 0
block: 78
itc : 28749645
LOOP: 0
'.b1_21'
block: 79
28749600
344995200
sysnsN.s
sysnsN.f90
Sql
IPC
Oracle Report
Lua Query
4114 (p16)ldfd f39=[r79],8 //0: {28:20} 1114
4115 (p16)ldfd f34=[r2],8 //0: {28:20} 1138
4116 (p17)fma.d f55=f7,f40,f49 ;; //16: {30:19} 1134
Static/dynamic details for sysnsN.s
Loop 2 Stats
Description
Static stats
Instrumentation stats
Floating-point bound 16
Estimated cycles 16*N + 32 Not executed
Estimated issues 16*N + 32
Issues / bound ratio 1
Store / load ratio 1
Fmas / load ratio 4
Integer / load ratio 0.37
Assembly line start 4109
Assembly line end 4267
Source line start 28
Oracle for line 28
Oracle results
Static stats for loop 3
Report
Unroll factor detected as 16 high degree of confidence
Based on FMAS and Store number unroll factor guessed to 16
Report
Scheduling is matching optimal bound
This is a clear sign of code quality but not a proof (beware of code bloating)
Report
Detected a loop version due to LOADS PAIR
Report
Vectorization opportunity
Numerous load/store may lead to bank conflict /store queue conflict/secondary miss
You should re-order array indices at the source code level
Report
Loop part of a 4-ways versioned loop
Loop ID : 5
Source Line : 28

CONCLUSION

- MAQAOPROFILE
  - Delivers precise diagnosis
  - Profiling scenarii
  - Enables user to choose the granularity of the information
  - Generates code tuning advices based on static and dynamic analyses
  - Guides optimization module
- Future work
  - Instrumentation of binary codes
  - Add other rules and metrics to performance adviser
  - Add other optimizations techniques
  - Port MAQAO to other architectures

Thank you